Lemma 5, *ab(t)* is the last STAR bridge on the enhanced forwarding path. Therefore, the enhanced forwarding path consists of three disjoint segments. The first segment, which is a tree path from *db(s)* to *ab(s)*, has a path length $d_T(db(s), ab(s))$. The second segment, which is an enhanced forwarding path from *ab(s)* to *ab(t)*, has a path length $d(ab(s), ab(t))$. The third segment, which is a tree path from *ab(t)* to *db(t)*, has a path length $d_T(ab(t), db(t))$. Therefore, $len(db(s), db(t))$, the length of the enhanced forwarding path from *db(s)* to *db(t)*, satisfies the following inequality.

$$\begin{aligned} len(db(s), db(t)) &= d_T(db(s), ab(s)) + d(ab(s), ab(t)) + d_T(ab(t), db(t)) \\ &\leq d_T(db(s), ab(s)) + d_T(ab(s), ab(t)) + d_T(ab(t), db(t)) \\ &= d_T(db(s), db(t)) \end{aligned}$$

Lemma 7:

A STAR forwarding path for a frame sent by an end station *s* to another end station *t* is a tree path if at least one of *ab(s)* and *ab(t)* is not defined.

Proof of Lemma 7:

FIGs. 24a-24d respectively show various exemplary scenarios for this lemma. By Lemma 2, if *ab(s)* is not defined, the STAR forwarding path must be a tree path because the frame will never be encapsulated. According to Case 9.2a of DF_STAR_Forwarding_Proc procedure, a frame will not be encapsulated unless *ab(t)* is defined.

Lemma 8:

When *ab(s)* and *ab(t)* are both defined, and *ab(s)* = *ab(t)*, then $len(db(s), db(t)) \leq d_T(db(s), db(t))$.

Proof of Lemma 8:

FIGs. 25a-25c show various exemplary scenarios for this lemma. When *ab(s)* = *ab(t)*, there are two cases: *ab(s)* = *db(s)* and *ab(s)* ≠ *db(s)*. In the first case, according to Case 8.2 of the ESL_Search_Proc procedure, *ab(s)* will send the frame according to the tree path, and the proof is complete. In the second case, if *db(t)* is an ancestor of *db(s)*, when the normal data frame travels upstream from *db(s)*, it will reach *db(t)* before *ab(s)* and so *ab(s)* won't receive that. In the second case, if *db(t)* is a descendant of *db(s)*, by Lemma 8, the STAR forwarding path must be a tree path, and the proof is complete.

Lemma 9:

When *ab(s)* and *ab(t)* are both defined, $ab(s) \neq ab(t)$, and *ab(s)* is an ancestor of *ab(t)*, then $len(db(s), db(t)) \leq d_T(db(s), db(t))$.

Proof of Lemma 9:

FIG. 26 shows an exemplary scenario for this lemma. When *ab(s)* is an ancestor of *ab(t)*, there are two cases: $ab(s) = db(s)$ and $ab(s) \neq db(s)$. In the first case, *ab(s)* may encapsulate the normal data frame and send it downstream along the tree path. When *ab(t)* receives an encapsulated data frame, it will strip off the header and trailer, and send the normal data frame according to the information in the FD. Therefore, the STAR forwarding path between *db(s)* and *db(t)* is a tree path. The proof is complete. When $ab(s) \neq db(s)$, *ab(s)* must be an ancestor of *db(s)* and *db(s)* must be an ancestor of *db(t)*.

Lemma 10:

When *ab(s)* and *ab(t)* are both defined, $ab(s) \neq ab(t)$, and *ab(s)* is a descendant of *ab(t)*, then $len(db(s), db(t)) \leq d_T(db(s), db(t))$.

Proof of Lemma 10:

FIG. 27 shows an exemplary scenario for this lemma. We have discussed the issue of the scenario in which *ab(t)* is an ancestor of *ab(s)* in Section V.D. When *ab(s)* finds out that *ab(t)* is an ancestor, it sends out the normal data frame on its root port. Since only *ab(s)* is allowed to encapsulate the normal data frame, all STAR bridges on the tree path between *ab(s)* and *ab(t)* will send the normal data frame on tree ports only. Then, the normal data frame will reach *db(t)* along the tree path without any redundant traversal and the proof is complete. All duplicate frames sent over the spanning tree will eventually be dropped.

We now present a theorem on the path length of the STAR bridge protocol.

Theorem 1 :

A STAR forwarding path for a frame sent by an end station *s* to another end station *t* leads from *db(s)* to *db(t)*, and has a path length *len(db(s), db(t))* that satisfies the following inequality.

$$len(db(s), db(t)) \leq d_T(db(s), db(t))$$

Proof of Theorem 1:

We first observe that any STAR forwarding path is either a tree path or an enhanced forwarding path. If it is a tree path, the proof is complete. Otherwise, we will show that the inequality still holds.

We divide all situations into the following scenarios:

1. *ab(s)* is not defined
2. *ab(s)* is defined
   - *2.1.* *ab(t)* is not defined
   - *2.2.* *ab(t)* is defined
     - *2.2.1.* *ab(s)* and *ab(t)* are on different branches
     - *2.2.2.* *ab(s)* and *ab(t)* are on the same branch

By Lemma 7, the STAR forwarding path is always a tree path except for the cases in scenario 2.2. By Lemma 6, the inequality holds for scenario 2.2.1. By Lemma 9 through Lemma 11, the inequality holds for scenario 2.2.2.

VIII. Conclusion

We have disclosed a new STAR Bridge Protocol that is backward compatible with the standard IEEE 802.1D Spanning Tree Bridge Protocol. The protocol described herein offers provably shorter alternate paths for frame forwarding whenever such paths are found, and uses tree paths otherwise. Incidentally, the standard protocol places a restriction on the maximum bridge diameter, which is the maximum number of bridges between any two points of attachments of end stations. The IEEE 802.1D specification recommends a maximum bridge diameter of 7. Given a set of LANs and bridges, there may not exist any single RST that could be built by the standard protocol to satisfy such a restriction. The STAR Bridge Protocol, on the other hand, may still satisfy the restriction because enhanced forwarding paths may reduce the bridge diameter. With the use of enhanced forwarding paths, frames that would have passed through the root bridge in the standard spanning tree may be diverted over alternate paths such that the load at the root bridge is likely to be alleviated.

Being backward compatible with the IEEE 802.1D standard, the proposed bridge protocol offers a smooth migration path to QoS-based bridging. Eventually, all old bridges will be phased out or replaced by STAR bridges. When there is no old bridge in the bridged network, distance vector routing can be used to find all-pair shortest paths. Since distance vectors are always forwarded one hop away, there will be no distant STAR neighbor when all bridges are STAR. Note that the IEEE 802.1D spanning tree is still needed as explained below.